

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

(повне найменування інституту, факультету)

Автоматизованих систем обробки інформації і управління

(повна назва кафедри)

«До захисту допущено»

В.о. завідувача кафедри

(підпис) О.А.Павлов
(ініціали, прізвище)

“ ” 2019 р.

Дипломний проект

на здобуття ступеня бакалавра

з напрямку підготовки 6.050103 «Програмна інженерія»

на тему Система моніторингу шлюбного польоту мурах

Виконав:

студент IV курсу, групи ІІІ-з51в

Клименко Данило Євгенович

(прізвище, ім'я, по батькові)

(підпис)

Керівник

асистент Коротенко А.А.

посада, науковий ступінь, вчене звання, прізвище, ініціали

(підпис)

**Консультант
з графічної
документації**

доц., к.т.н., Ліщук К.І.

посада, науковий ступінь, вчене звання, прізвище, ініціали

(підпис)

Рецензент:

доц., к.т.н., доц. кафедри ТК Ткач М.М.

посада, науковий ступінь, вчене звання, прізвище, ініціали

(підпис)

Засвідчую, що у цьому дипломному проекті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

Пояснювальна записка до дипломного проекту

на тему: *Система моніторингу шлюбного польоту мурах*

Київ – 2019 року

АНОТАЦІЯ

До бакалаврського дипломного проекту Клименко Данила Євгеновича

На тему: «Система моніторингу шлюбного польоту мурах»

Серед людей, що займаються спостереженням за мурахами, існує проблема відсутності місця де буде достатня кількість достовірних даних щодо тенденцій шлюбного польоту мурах для певної території невеликого порядку, що суттєво ускладнює визначення календарних рамок польоту для конкретного виду на даній або сусідньої території з достатньо низьким відхиленням.

Мета дипломного проекту: централізувати розміщення достовірних повідомлень про шлюбний політ мурах.

Пояснювальна записка дипломного проекту складається з чотирьох розділів, містить 26 таблиць, 4 додатки та 11 джерел - загалом 51 сторінок.

У першому розділі була проаналізована предметна область та існуючі рішення, виділені функціональні та нефункціональні вимоги.

У другому розділі було спроектовано програмне забезпечення.

У третьому розділі проведено тестування програмного забезпечення за розробленим планом тестування. Описано процес тестування.

В четвертому розділі описано розгортання та впровадження програмного забезпечення.

КЛЮЧОВІ СЛОВА: ШЛЮБНИЙ ПОЛІТ МУРАХ , МОНІТОРИНГ, ASP.NET CORE

					КПІ.ІП-з4104.045490.02.81	Арк.
						2
Змн.	Арк.	№ докум.	Підпис	Дата		

ABSTRACT

To the Bachelor's thesis of Klymenko Danylo

On a subject: «Ants Nuptial Flight Monitoring Software»

The explanatory note of the diploma project consists of four sections, contains 26 tables, 4 annexes and 11 sources - total 51 pages.

The aim of the diploma project: to facilitate the search and selection of a gift for a particular person by analyzing the profile of a social network..

In the first section the subject area was considered. An application from the business point of vision is analyzed. Requirements were determined.

In the second section, the architecture of the software was developed. A structural diagram of classes and a database are constructed.

In the third section, the software was tested according to the developed test plan. The process of testing described.

The fourth section describes the deployment and implementation of the software.

KEYWORDS: ANT NUPTIAL FLIGHT, MONITORING , ASP.NET CORE

					КПІ.ІП-34104.045490.02.81	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКРОЧЕНЬ І ТЕРМІНІВ.....	5
ВСТУП.....	6
1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	7
1.1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	7
1.2 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ	9
1.3 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	15
1.4 Висновки до розділу	23
2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	24
2.1 МОДЕЛЮВАННЯ ТА АНАЛІЗ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	24
2.2 АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	25
2.3 СТРУКТУРА ФАЙЛІВ ПРОЕКТУ	27
2.4 БАЗА ДАНИХ	32
2.5 КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	36
2.6 Висновки по розділу	42
3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	43
3.1 АНАЛІЗ ЯКОСТІ ПЗ.....	43
3.2 ОПИС ПРОЦЕСІВ ТЕСТУВАННЯ.....	43
3.3 ОПИС КОНТРОЛЬНОГО ПРИКЛАДУ	44
3.4 Висновки по розділу	48
4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	49
4.1 РОЗГОРТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	49
ВИСНОВКИ	50
ПЕРЕЛІК ПОСИЛАНЬ.....	51

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Шлюбний політ – важлива фаза відтворення більшості мурашок, термітів та деяких видів бджіл.

БД – база даних.

ПЗ – програмне забезпечення.

ASP.NET Core – технологія створення веб-застосунків і веб-сервісів від компанії Майкрософт.

MVC – архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення.

EntityFramework Core – об'єктно-орієнтована технологія доступу до даних.

CodeFirst – підхід до побудови БД, в якому спочатку описуються моделі, з яких за допомогою генеруються скрипти створення та міграції бази даних.

Razor – двигун генерації html коду із C# коду, який використовується в ASP.NET фреймворку.

SOLID – абревіатура складена з перших літер п'яти базових принципів об'єктно-орієнтованого програмування та дизайну запропонована Робертом Мартіном.

DDD - підхід до моделювання складного об'єктно-орієнтованого програмного забезпечення.

Web - інтернет-простір.

API – програмний інтерфейс.

					КПІ.ІП-34104.045490.02.81	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

У світі існує багато різних хобі, одне з них – спостереження за мурахами. Люди які займаються цим хобі називаються мірмікипери. Звичайно, окрім спільного хобі їх об'єднує ще спільні проблеми. Одна з таких проблем – дуже складно визначити календарні рамки з низьким відхиленням для шлюбного польоту на певній території певного виду. Це зумовлено принципами варіативності календарних рамок в залежності від глобальних та локальних погодних умов поточного і минулого року. На даний момент це вирішуються принципом передачі інформації “з вуст в уста” на форумах та у відповідних групах в соціальних мережах. На форумах інформація достовірна, але її мало, у соціальних мережах більше інформації, але її ступінь достовірності досить низька. Обидва джерела інформації мають недолік у відсутності функціоналу фільтрації повідомлень за таксоном і локацією польоту. Також для визначення тенденцій потрібну вручну збирати інформацію.

Дану ситуацію можна поліпшити шляхом розробки централізованої системи фіксації та верифікування повідомлень про шлюбні польоти. З можливістю фільтрації даних за таксоном і локацією, візуалізацією цих даних, що і являє собою даний проект.

					КПІ.ІП-34104.045490.02.81	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Аналіз предметної області

Одне з не найбільш популярних, але від цього не менш цікаве захоплення – спостереження за життям і розвитком колонії мурах. Для цього потрібна колонія мурах. Отримати колонію можна двома шляхами: купити або спіймати запліднену королеву (або купити колонію з заплідненою королевою). Так як в штучних умовах отримати запліднену королеву можна сказати неможливо, продавці їх ловлять або перепродають. Зловити запліднену королеву можливо у найближчі 24 годин після початку шлюбного польоту.

Приклад із власних спостережень. Через 1 годину після початку шлюбного польоту мурах виду *Lasius niger* на відстані 5 метрів від найближчого гнізда за 1 хвилину на одному місці було нараховано 74 запліднені королеви. На наступний день (приблизно 24 години після польоту) за півтора години прогулянки біля місць де знаходяться їх гнізда було нараховано 23 запліднені королеви. Ще через день (приблизно 48 годин після початку польоту) за півтора години було знайдено 3.

Даний приклад гарно ілюструє шанси спіймати запліднену королеву, але треба мати на увазі, що даний вид – один з найбільш розповсюджених видів на території Європи з помірним кліматом, гарно пристосований до умов міста і шлюбний політ характерний дуже великою кількістю статевих особин, тобто можна просто вийти з дому і випадково натрапити на королеву не готуючись і не шукаючи їх спеціально. Саме з цих причин це один з видів який з легкістю можна купити.

Зазвичай види, які не проживають на території міста або/і кількість статевих особин які вилітають з одного гнізда під час шлюбного польоту, невелика - більш бажанні, цікаві і як наслідок дорожчі, якщо взагалі є у продажу. За тими видами, що проживають за територією проживання людини треба спеціально добиратися до ареалу їх проживання.

					КПІ.ІП-34104.045490.02.81	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

Серед чинників, що впливають на шанси піймати королеву, але які ми не можемо вплинути можна виділити:

- масштабу шлюбного польоту;
- характер місцевості;
- розмір королеви.

Єдиний чинник на який ми можемо вплинути, але який має не найменше значення – визначення календарних рамок шлюбного польоту з мінімальним відхиленням.

Чинники що впливають на календарні рамки:

а) сезонні:

- 1) погодні умови поточно сезону;
- 2) погодні умови минулого сезону;

б) постійні:

- 1) видова приналежність;
- 2) специфіка погодних умов території.

Для визначення календарних рамок з мінімальним відхиленням потрібна система яка дозволяє фіксувати факт шлюбного польоту, на основі зібраних даних передбачає нові на основі даних метеорологічних станцій та повідомляє зацікавлених учасників. Згідно сформульованих вимог можна виділити декілька досить великих прецедентів ідеальної системи, спрощена схема якої наведено на рис. 1.1.



Рисунок 1.1 - Спрощена схема прецедентів ідеальної системи

Так як ідеальна система виходить за рамки дипломного проекту, у цій роботі буде розглянута саме частина фіксації повідомлень про шлюбний польотів мурах так як від роботи цієї частини залежать інші.

Важливою частиною процесу фіксації повідомлення являю собою його ідентифікація таксону мурахи. Зазвичай для визначення таксону використовують визначники, але елементи які там описано зазвичай неможливо помітити без застосування спеціальної техніки. Наприклад підрід *Raptiformica* відрізняється від інших підродів *Formica* наявністю ямки біля жвал (див. Рис. 1.2), ширина голови яких приблизно 1мм.



Рисунок 1.2 – Різниця між *Raptiformica* та іншими підродами *Formica*.

Raptiformica праворуч, іншу ліворуч.

Для такого способу визначення таксону потрібно непогані навички фотографії і технічне забезпечення. Зазвичай у людей немає всього необхідного тому фотографії доповнюють інформацією про місцевість, яке дуже допомагає у визначенні виду. Характер місцевості не зазначено у визначниках, тому недовічним мірмікіперам допомагають більш досвідчені.

1.2 Аналіз існуючих рішень

На даний момент отримати данні про шлюбні польоти можна з декількох джерел:

- спеціалізовані форуми;
- групи у соціальних мережах;
- наукова література.

У кожного з джерел є свої переваги і недоліки, які розглянемо далі.

1.2.1 Форуми

Взагалі теми пов'язані за шлюбним польотом – норма, одиночні так і централізовані. На одиночних темах люди питають питання на які вже відповідали більш досвідчені учасники клубу, проте на форумі інколи досить важко відшукати ці повідомлення. Також бувають питання про специфічні території. Подібне питання на форумі antflight.org зображено на рис. 1.3

Новости о начале лёта муравьев по регионам.

Пнд, 2010-04-12 20:43 — Komandos

К сожалению в Интернете бываю чаще чем на улице, поэтому не хочется пропустить начало лета муравьев в своем регионе.

Если кто из Киевского или другого региона станет свидетелем начала лета любого вида муравьев – просьба кинуть сюда информацию.

Рисунок 1.3 - Приклад питання про календарні рамки шлюбного польоту.

Доступний з <<http://antclub.org/node/3041>>

У таких повідомлень є одна характерна риса – відшукати ці повідомлення дуже важко.

Звичайно деякі учасники форуму намагаються централізувати цю інформацію.

На antflight.org існує тема, присвячена централізації повідомлень про шлюбний політ. Нажаль поповнюється вона аби як, повідомлень для проведення статистики дуже мало, відсутня система пошуку по повідомленням, системи фільтрації, проте ці повідомлення досить точні так як їх можуть залишати зареєстровані учасники, а процес реєстрації має пункт лист-рекомендації без якого неможливо стати учасником клубу.

Форум antclub.ru надає можливість додавати повідомлення про шлюбні польоти, та відображає їх у вигляді таблиці (див. Рис.1.4)

					КПІ.ІП-з4104.045490.02.81	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

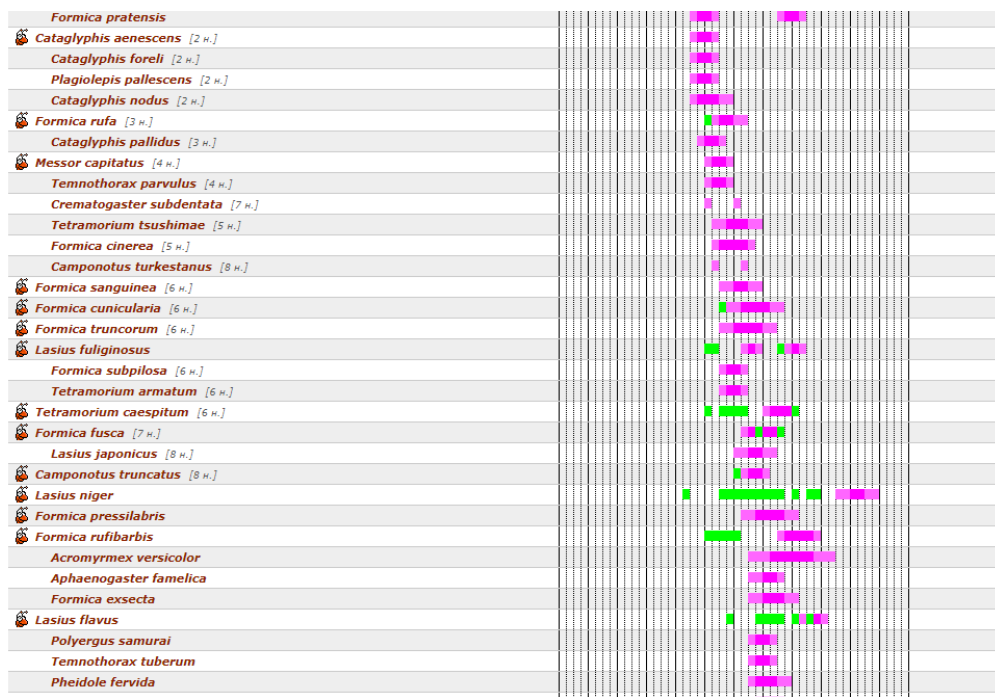


Рисунок 1.4 - Таблиця шлюбних польотів на сайті antflight.ru. Світлим тоном зображено наявність повідомлень про шлюбний політ у вказаний період часу. Доступна з <<http://antclub.ru/flight>>

На жаль дане рішення має ряд мінусів зразу у декількох площинах:

- дизайн таблиць та самого ресурсу не підтримує мобільні версії;
- відсутня можливість фільтрації даних за локацією польоту, року, таксономією;
- валідність даних. Для можливості додавання нового повідомлення про політ треба бути лише зареєстрованим користувачем (так як визначення виду потребує певної кваліфікації реєстрація не є достатнім рівнем підтвердження достовірності точності цього повідомлення).

Форум www.formiculture.com пропонує подібні таблиці , що зображено на Рис. 1.5

Ant Mating Chart (Global)

Species	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
Atta												
Atta texana				1	9	3						
Brachymyrmex												
Brachymyrmex depilis			1	1			2	10	5	3		
Brachymyrmex obscurior						1						
Brachymyrmex patagonicus				1	6	2	2	1	1	1		
Brachymyrmex sp.			1	1	2	2	3	6	5			
Brachyponera												
Brachyponera chinensis					1	4	1					
Camponotus												
Camponotus aethiops						1	1	1				
Camponotus americanus			5	5	4	1		2		1		
Camponotus anthrax					1							
Camponotus auriventris				1								
Camponotus caryae				1	1							
Camponotus castaneus			4	4	29	29	2					
Camponotus chromaiodes			4	17	9	1	1					
Camponotus consobrinus												2

Рисунок 1.5 - Скріншот таблиці шлюбних польотів на веб-ресурсі

[formiculture.com](http://www.formiculture.com). Таблиця доступна з посилання

<<http://www.formiculture.com/index.php/topic/181-ant-mating-chart/>>

У даного рішення ті ж самі проблеми, що й у ресурсу antclub.ru. Ще одне подібне локальне за своїми даними рішення пропонує веб-ресурс ameisenforum.de, зображено на Рис. 1.6.

Art (kursiv = Exotisch)	Jan.	Feb.	Mrz.	Apr.	Mai	Juni	Juli	Aug.	Sept.	Okt.	Nov.	Dez.	Bemerkung*
<i>Acromyrmex versicolor</i>													
<i>Anergates atratulus</i>													
<i>Aphaenogaster subterranea</i>													
<i>Aphaenogaster tennesseensis</i>													
<i>Brachymyrmex depilis</i>													
<i>Camponotus aethiops</i>													
<i>Camponotus americanus</i>													
<i>Camponotus cruentatus</i>													
<i>Camponotus fallax</i>													w 16-17
<i>Camponotus herculeanus</i>													m 17-19
<i>Camponotus lateralis</i>													
<i>Camponotus ligniperda</i>													w 15-18
<i>Camponotus noveboracensis</i>													
<i>Camponotus pennsylvanicus</i>													
<i>Camponotus piceus</i>													
<i>Camponotus substitutus</i>													
<i>Camponotus truncatus</i>													
<i>Camponotus vagus</i>													w 13-15
<i>Cataglyphis nodus</i>													
<i>Cerapachys augustae</i>													
<i>Crematogaster cerasi</i>													
<i>Crematogaster scutellaris</i>													
<i>Diacamma rugosum</i>													
<i>Dolichoderus quadripunctatus</i>													
<i>Formica aquilonia</i>													
<i>Formica bruni</i>													
<i>Formica cinerea</i>													
<i>Formica cunicularia</i>													m 9-11
<i>Formica exsecta</i>													
<i>Formica foreli</i>													
<i>Formica forsslundi</i>													
<i>Formica fusca</i>													
<i>Formica gagates</i>													
<i>Formica glauca</i>													
<i>Formica lefrancoisi</i>													
<i>Formica lemani</i>													

Рисунок 1.6 - Скріншот таблиці шлюбних польотів на веб-ресурсі

ameisenforum.de. Таблиця доступна з посилання

<<http://ameisenforum.de/schwarmflug.php>>

1.2.2 Групи у соціальних мережах

За допомогою груп у соціальних мережах можна простежити потрібну інформацію у постах, але цей спосіб має ряд недоліків:

- відсутні фільтри за локацією та таксоном, через що пошук потрібних повідомлень неймовірно повільний;
- відсутня верифікація, через що недосвідченому мірмікіперу буде неймовірно важко відшукати потрібну інформацію.

Звичайно поряд з недоліками є й переваги:

- повідомлень більше порівняно з темою на форумі;
- повідомлення часто більш конкретні у плані інформації про локацію.

Приклад повідомлення про шлюбний політ у соціальних мережах наведено на рис. 1.7.

Winter ants are having their nuptial flights. They usually fly on days that are 70F - 75F around 4 pm till 8pm. -Nook



Рисунок 1.7 – Приклад повідомлення про шлюбний політ у соціальній мережі. Повідомлення доступне за посиланням

<<https://www.facebook.com/NebraskaAnts/posts/2229590744036915>>[11]

Як вже було зазначено вище визначити вид мурахи що зображена на рис. 1.7 майже неможливо через недостатню деталізацію на фотографії.

1.2.3 Наукова література

Що до достовірності інформації викладеній у науковій не потрібно переживати але недоліки теж є :

- інформація про календарні рамки викладена з дуже великим відхиленням (+- 3тиждні);
- книжки зазвичай присвячені певному роду\підроду і певній території порядку країни;
- данні про календарні рамки відносно рідкісна інформація, так як вивчення мурах відноситься до загальних наук;
- календарні рамки зазначаються для локацій занадто великого порядку, що сильно збільшує відхилення;
- дослідження проводяться не про календарні рамки, а про інші фактори шлюбного польоту, що робить календарні рамки побічною інформацією.

1.3 Аналіз вимог до програмного забезпечення

Акторами системи є: неавторизований користувач, користувач.

Визначимо, які дії або варіанти використання вони виконують в системі, для цього наведемо таблицю 1.1, в якій описані актори, варіанти використання та їх описи дій.

Таблиця 1.1 – Варіанти використання

Актор	Варіант використання	Опис дії варіанта використання
Гість	Реєстрація	Неавторизованому користувачу надається форма для реєстрації
Гість	Перегляд тенденцій польотів	Гостю надається можливість перегляти тенденції шлюбних польотів мурах
Користувач	Автентифікація	Користувачу надається можливість пройти автентифікацію шляхом введення пари логіну - паролю

Продовження таблиці 1.1

Користувач	Ведення своїх повідомлень про шлюбний політ мурах	Користувачу надається можливість додати , змінювати , видаляти свої повідомлення про шлюбний політ.
Користувач	Запросити верифікацію повідомлення	Користувачу надається можливість запросити верифікацію свого повідомлення
Модератор	Верифікація повідомлень про шлюбний політ	Модератору надається можливість модерації повідомлень про шлюбний політ , котрі були відіслані с прохонанням верифікації

Відповідно до визначених варіантів використання побудовано загальну модель варіантів використання, яка наведена на Рис. 1.8

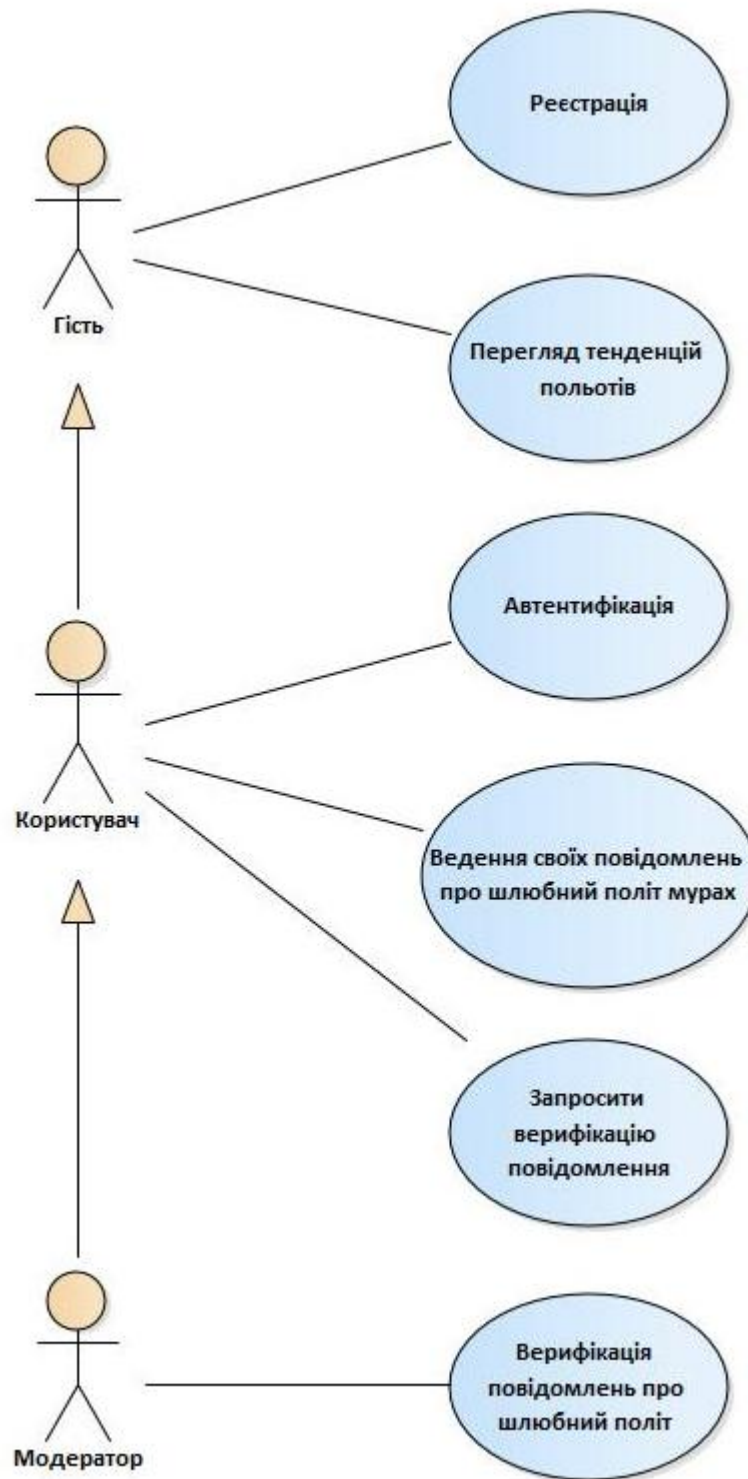


Рисунок 1.8 – Діаграма варіантів використання (Use Case Diagram)

1.3.1 Розроблення функціональних вимог

Відповідно до визначених варіантів використання виявлено функціональні вимоги та встановлено їхній пріоритет, результат наведено у таблиці 1.2.

Таблиця 1.2 – Виявлені вимоги з варіантів використання

Актор	Варіант використання	Функціональна вимога	Пріоритет
Гість	Реєстрація	1.1) Система здійснює перевірку наявності користувача з такою електронною адресою	Високий
		1.2) Система перевіряє надійність паролю	Високий
		1.3) Система додає нового користувача	Високий
Гість	Перегляд тенденцій польотів	2.1) Система полегшує процес пошуку потрібного таксону	Середній
		2.2) Система полегшує процес вибору локації	Низький
		2.3) Система відображає тенденції шлюбного польоту	Високий

Продовження таблиці 1.2

Користувач	Автентифікація	3.1) Система здійснює перевірку наявності користувача з паролю логін -пароль	Високий Високий
Користувач	Ведення своїх повідомлень про шлюбний політ мурах	4.1) Система надає можливість створювати повідомлення про шлюбний політ. 4.2) Система надає можливість редагувати створені повідомлення про шлюбний політ. 4.3) Система надає можливість видаляти повідомлення про шлюбний політ.	Високий Високий Високий
Користувач	Запросити верифікацію повідомлення	5.1) Система надає можливість запросити верифікацію створеного повідомлення	Високий

Продовження таблиці 1.2

Модератор	Верифікація повідомлень про шлюбний політ	6.1) Система надає можливість відмовляти у верифікації повідомлення про шлюбний політ	Високий
		6.1.1) Система надає можливість надати пояснення відмови	Високий
		6.2) Система надає можливість верифікувати повідомлення про шлюбний політ	Високий

Відповідно виявлених вимог побудовано загальну модель вимог, що зображена на рис.1.9.

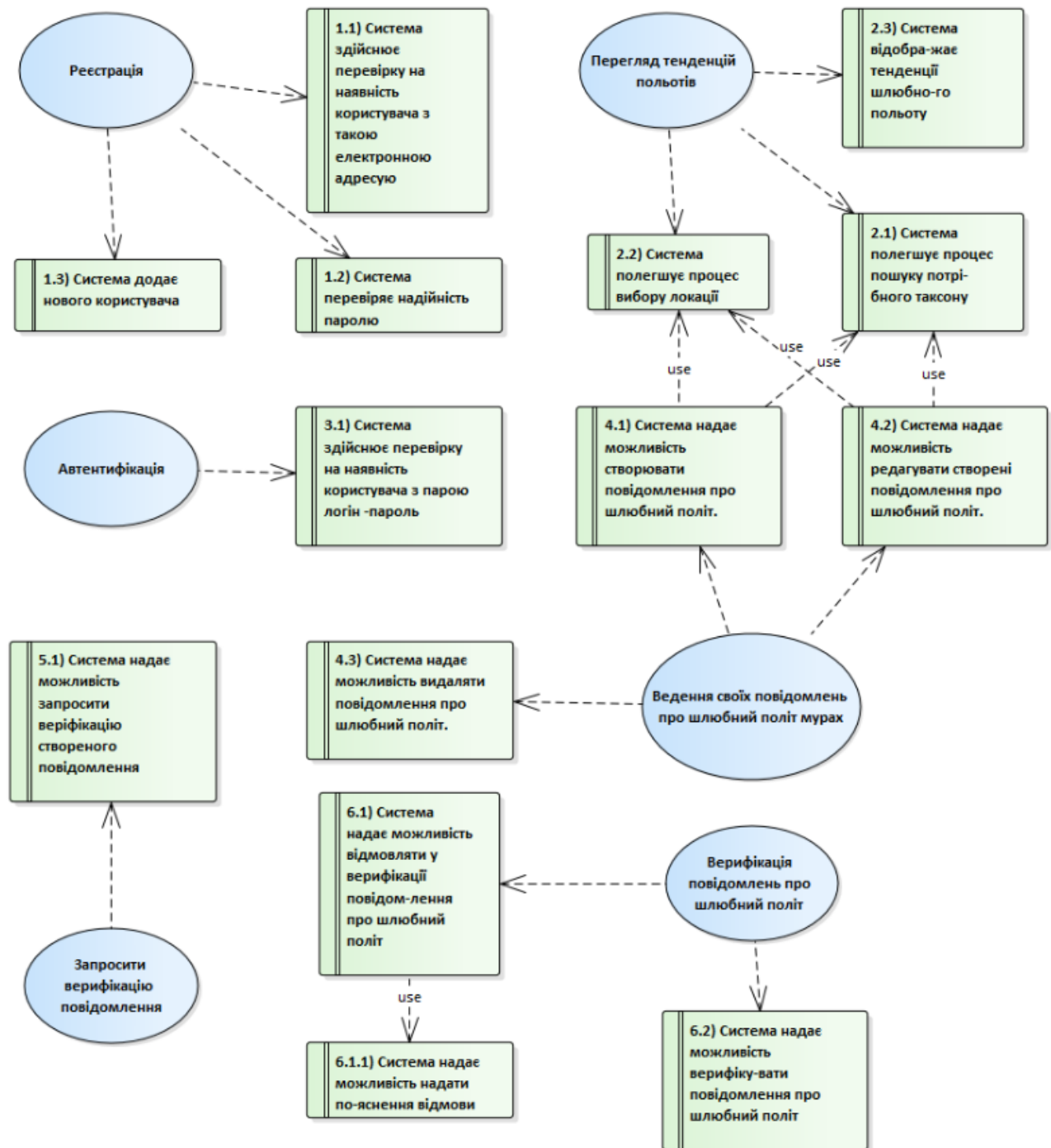


Рисунок 1.9 - Модель функціональних вимог

1.3.2 Розроблення нефункціональних вимог

- підвантаження даних не повинно блокувати екран;
- дизайн має бути адаптивним до розміру екрану - підтримувати мобільну та десктопну версію;
- мова додатку – російська.

1.3.3 Постановка комплексу завдань для розробки

Розробка призначена для централізації повідомлень про шлюбний політ мурах.

Метою створення програмного забезпечення є полегшення пошуку інформації про тенденції шлюбного польоту мурах.

На основі аналізу предметної області, аналізу існуючих програмних рішень, варіантів використання та розроблених функціональних та нефункціональних вимог до програмного забезпечення сформовано наступні задачі розробки:

- авторизація в системі;
- ведення повідомлень про шлюбні польоти мурах;
- верифікація повідомлень;
- відображення тенденцій шлюбного польоту мурах.

1.4 Висновки до розділу

У першому розділі була викладено інформацію про предметну область, її проблеми, що можуть бути вирішені за рахунок системи автоматизації, проаналізовані варіанти використання та виставлені функціональні та нефункціональні вимоги. Порівняльна характеристика з ідеальною системою та існуючими рішеннями зображено на рис.1.10

					КПІ.ІП-34104.045490.02.81	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

		Форум		Повідомлення в соціальній мережі	Ідеальна система фіксації	Даний дипломний проект
		Таблиця	Спеціальна тема з повідомленнями			
Фільтрація повідомлень						
	За локацією	-	-	-	+	+
	За таксономією	-	-	-	+	+
	За календарними рамками	-	-	-	+	+
Верифікація повідомлень		-	+/-	-	+	+
Можливість продивитись конкретне повідомлення		+/-	+	+	+	+/-
Не потрібно самостійно проводити аналітику		+	-	-	+	+

Рисунок.1.10 - Порівняльна характеристика дипломного проекту, ідеальною системою та існуючими рішеннями

Було створено діаграму варіантів використання, розроблено систему функціональних і нефункціональних вимог, результати подані в моделі функціональних вимог і на матриці взаємозв'язку вимог з варіантами використання.

При аналізі існуючих рішень було виявлено, що існуюча системи мають суттєві недоліки, які можна вирішити.

2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Моделювання та аналіз програмного забезпечення

Програмне забезпечення включає наступні обмежені контексти:

- автентифікація та авторизація;
- локації;
- повідомлення про шлюбний політ.

Серед контексту повідомлення виділені такі агрегати :

- повідомлення про шлюбний політ;
- таксон.

Центральним звичайно є повідомлення про шлюбний політ “Nuptial flight message”. Повідомлення проходить через декілька станів що зображені разом з переходами на діаграмі станів (рис 2.1).

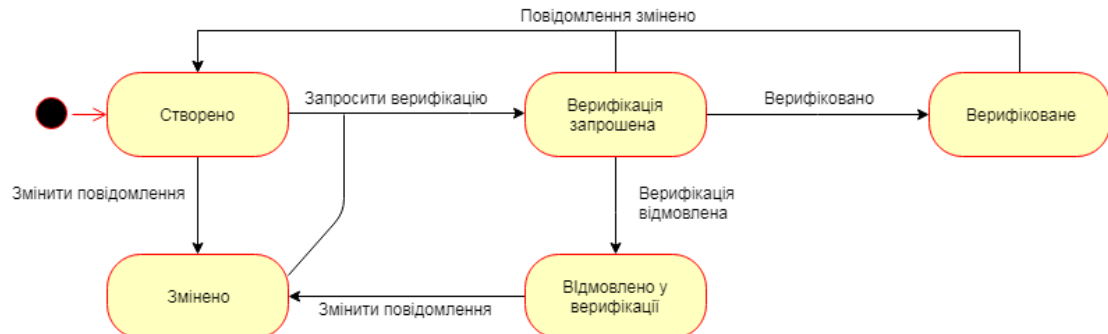


Рисунок 2.1 Діаграма станів повідомлення про шлюбний політ

Згідно с рис 2.1 виділено наступні стани :

- створено;
- верифікація запрошена;
- очікуються зміни;
- верифіковане.

Склад повідомлення наведено у табл. 2.1

Таблиця 2.1 – Опис суб'єкта Nuptial flight message

Ключ	Значення
Id	Унікальний ідентифікатор повідомлення
Creator Id	Унікальний ідентифікатор користувача, що створив повідомлення
State History	Історія змін повідомлення станів повідомлення
Nuptial flight	Інформація про шлюбний політ

2.2 Архітектура програмного забезпечення

При проектуванні системи були ураховані принципи SOLID, DDD , CQRS, була використана багатошарова архітектура яка зображена на рис 2.2 зображено її частини , та їх взаємодія

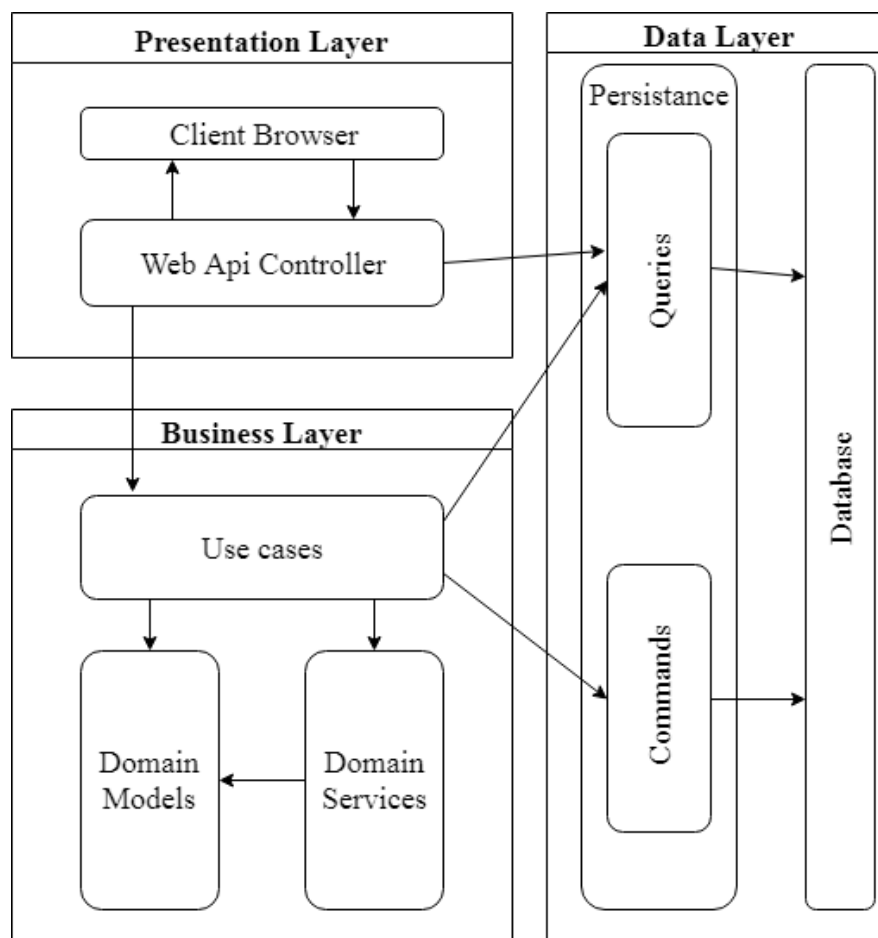


Рисунок. 2.2 - Елементи архітектури та їх взаємодія.

2.2.1 Presentation layer

Цей шар на даний момент складається лише з Web API, але може розширюватись у напрямку мобільних клієнтів, чат ботів. Як частина, відповідає за презентацію. В обов'язки Web API входить:

- створювати HTML сторінки;
- роутінг;
- мапінг входячих даних у відповідні об'єкти Business Layer;
- формування об'єктів представлення ;
- валідація даних, які не пов'язані з бізнес логікою, але потрібні для цілісності об'єктів бізнес логіки.

2.2.2 Business Logic Layer

Цей шар відповідає за логіку предметної області. Він складається з трьох частин: use cases, domain models , domain services.

Domain models являють собою об'єкти які описують предметну область, складаються з суб'єктів, агрегатів, та подій, які були проведені в агрегаті. Агрегати знають про свої суб'єкти і не знають нічого про інші агрегати, окрім їх унікального ідентифікатора.

Domain services об'єкти які проводять взаємодію між агрегатами, якщо потрібно.

Use cases – об'єкти, що є проміжною точкою поміж presentation layer, data layer і business layer, віддзеркалюють варіанти використання програмного забезпечення. Носять функції:

- за допомогою запитів до шару даних відтворює потрібні об'єкти предметної області;
- викликає методи, які відповідають процесу обробки даного варіанту використання;
- сповіщає систему зберігання, що операція закінчена і об'єкти можуть бути збережені;

					КПІ.ІП-з4104.045490.02.81	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

— сповіщає поточну та інші системи про результат виконання варіанту використання.

2.2.3 Data Layer

Відповідає за збереження інформації складається з двох елементів: бази даних і системи мапінгу таблиць у об'єкти і навпаки та трекінгу змінень цих самих об'єктів.

Система трекінгу и мапінгу об'єктів предметної області, умовно, за допомогою інтерфейсів, розділена на запити і команди. Команди додають, обновлюють і видаляють об'єкти у базі даних. Запити відтворюють об'єкти предметної області.

2.3 Структура файлів проекту

Проект складається з декількох рішень, які згруповані по папкам. Назва папок та пояснення наведені у табл. 2.2

Назва папки	Пояснення
Clients	Папка з клієнтами. На даний момент це лише веб-клієнт
Identity	Папка з рішеннями що до авторизації та автентифікації
Location	Папка з рішеннями що до визначення локації
NuptialFlights	Папка з рішеннями що до шлюбного польоту мурах
Persistence	Папка з рішеннями що до зберігання інформації

На рисунку 2.3 зображена структура файлів

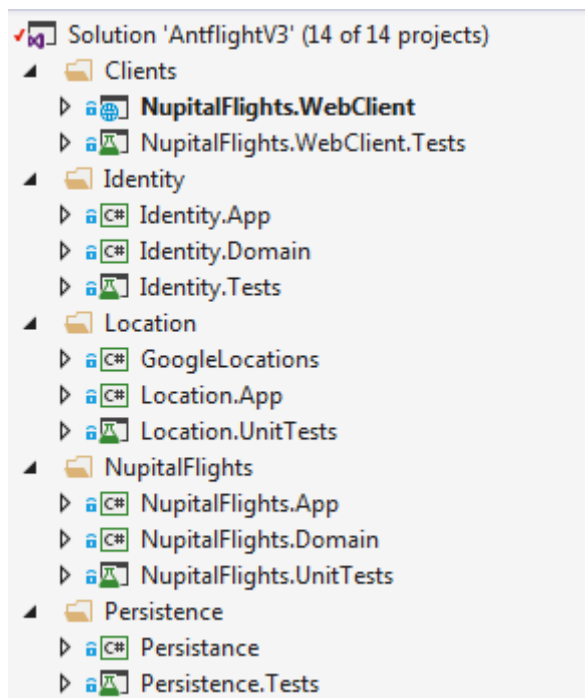


Рисунок 2.3 – Структура файлів проекту

Згідно з малюнку можна побачити що проект розбито на частини відповідно до озвучених вище окремих контекстів та архітектури. Кожний окремий контекст , якщо це шар бізнес логіки, або шар архітектури має окремий проект, в якому знаходяться тести.

На рисунку 2.4 зображена структура веб-клієнту додатку.

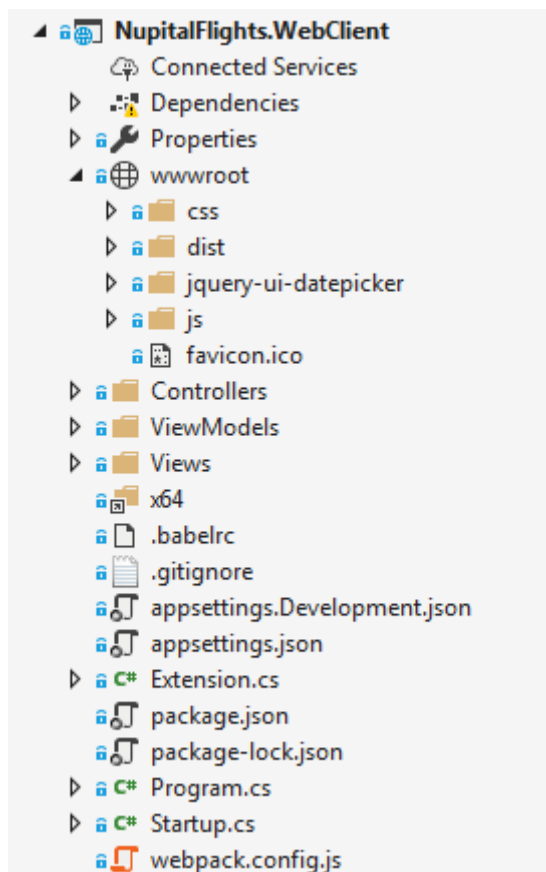


Рисунок 2.4 – Структура рішення веб клієнту

Рішення веб клієнту складається з:

- папки `wwwroot`, в якій знаходяться дані, які публічно доступні при розгортанні додатку;
- папки `Controllers`;
- папки `ViewModels`, об'єкти, що надходять до контроллерів;
- папки `View`, в якій знаходяться представлення, з яких Razor формує HTML сторінки.

2.3.1 Опис класів програмного забезпечення

Загальну схему класів предметної області наведено в документі *КПІ.ІП-34104.045490.10.99 СС «Схема структурна класів програмного забезпечення»*

Таблиця 2.3 – Опис класів Domain Models повідомлень про шлюбний політ

Назва класу	Тип	Опис
NupitalFlightMessage	Агрегат	Повідомлення про шлюбний політ
NupitalFlight	Value object	Опис повідомлення про шлюбний політ
MessageState	Value object	Базовий клас стану повідомлення.
Created	Value object	Стан повідомлення. Створено
AskedForVerification	Value object	Стан повідомлення. Запрошено верифікацію
Verified	Value object	Стан повідомлення. Верифіковано
VerificationRejected	Value object	Стан повідомлення. Відмовлено у верифікації
Edited	Value object	Стан повідомлення. Змінено
Taxon	Агрегат	Таксон

Таблиця 2.4 – Опис методів класу NupitalFlightMessage

Метод	Вхідні параметри	Опис
UpdateNupitalFlight	NupitalFlight nupitalFlight	Змінює інформацію про політ
Verify	int moderatorId	Приймає верифікацію повідомлення
RejectVerification	int moderatorId, string rejectReason	Відмова у верифікації
AskVerification		Запросити верифікацію

Таблиця 2.5 – Опис класів-команд предметної області

Команда	Вхідні параметри	Опис
CreateNupitalFlightCommand	int creatorId, NupitalFlight nupitalFlight	Створити повідомлення про шлюбний політ
DeleteMessageCommand	int messageId	Видалити повідомлення
EditNupitalFlightCommand	int messageId, NupitalFlight nupitalFlight	Змінити повідомлення про шлюбний політ
AskVerification	int messageId	Запросити верифікацію
CancelVerificationRequestCommand	int messageId	Відмінити запит на верифікацію
RejectFlightMessageCommand	int messageId, int moderatorId, string reason	Відмовити у верифікації
VerifyNupitalFlightCommand	int moderatorId, int messageId	Верифікувати повідомлення про шлюбний політ

Таблиця 2.6 – Опис обробників команд до предметної області

Обробник	Команда	Вхідні параметри
CreateNupitalFlightCommandHandler	CreateNupitalFlightCommand	IWrite<NupitalFlightMessage> writer
DeleteMessageCommandHandler	DeleteMessageCommand	INupitalFlightsQueries queries, IWrite<NupitalFlightMessage> writer

Продовження таблиці 2.6

EditNupitalFlight CommandHandler	EditNupitalFlight Command	INupitalFlightsQueries nupitalFlightsQueries, IWrite<NupitalFlightMessage> writer
AskNupitalFlight Verification CommandHandler	AskVerification	INupitalFlightsQueries nupitalFlightsQueries, IWrite<NupitalFlightMessage> writer
CancelVerification Request CommandHandler	CancelVerificationRequest Command	INupitalFlightsQueries queries, IWrite<NupitalFlightMessage> write
RejectFlightMessage CommandHandler	RejectFlightMessage Command	INupitalFlightsQueries queries, IWrite<NupitalFlightMessage> write
VerifyNupitalFlight CommandHandler	VerifyNupitalFlight Command	INupitalFlightsQueries queries, IWrite<NupitalFlightMessage> write

2.4 База даних

Загальна схема бази даних зображено на схемі (Документ – *КПІ.ІП-34104.045490.08.99.СБД «Схема бази даних»*)

Таблиця 2.7 - Опис таблиць бази даних

Таблиця	Опис
asp_net_users	Користувачі
asp_net_user_roles	Ролі
asp_net_roles	Ролі користувачів

messages

Повідомлення про шлюбний політ

Продовження таблиці 2.7

messages_states	Стани повідомлень
taxons	Таксони
migrations	Міграції бази даних

Таблиця 2.8 - Таблиця migrations

Ім'я колонки	Тип даних	Not Null
ProductVersion	pg_catalog.varchar	true
MigrationId	pg_catalog.varchar	true

Таблиця 2.9 - Таблиця messages

Ім'я колонки	Тип даних	Not Null
id	pg_catalog.int4	true
creator_id	pg_catalog.int4	true
description	pg_catalog.text	false
location_id	pg_catalog.text	false
taxon_id	pg_catalog.int4	true
timestamp	pg_catalog.timestampz	true

Таблиця 2.10 - Таблиця messages_states

Ім'я колонки	Тип даних	Not Null
id	pg_catalog.int4	false
timestamp	pg_catalog.timestampz	true
message_id	pg_catalog.int4	false
state_code	pg_catalog.int2	true
moderator_id	pg_catalog.int4	false
reason	pg_catalog.text	false

verificator_id	pg_catalog.int4	false
----------------	-----------------	-------

Таблиця 2.11 - Таблиця taxons

Ім'я колонки	Тип даних	Not Null
id	pg_catalog.int4	true
level	pg_catalog.int4	true
full_name	pg_catalog.text	false
search_name	pg_catalog.text	false
parent_id	pg_catalog.int4	false

Таблиця 2.12 - Таблиця asp_net_users

Ім'я колонки	Тип даних	Not Null
id	pg_catalog.int4	true
user_name	pg_catalog varchar	false
normalized_user_name	pg_catalog varchar	false
email	pg_catalog varchar	false
normalized_email	pg_catalog varchar	false
email_confirmed	pg_catalog.bool	true
password_hash	pg_catalog.text	false
security_stamp	pg_catalog.text	false
concurrency_stamp	pg_catalog.text	false
phone_number	pg_catalog.text	false
phone_number_confirmed	pg_catalog.bool	true
two_factor_enabled	pg_catalog.bool	true
lockout_end	pg_catalog.timestampz	false
lockout_enabled	pg_catalog.bool	true
access_failed_count	pg_catalog.int4	true

Таблиця 2.13 - Таблиця asp_net_user_tokens

Ім'я колонки	Тип даних	Not Null
user_id	pg_catalog.int4	true
login_provider	pg_catalog.text	true
Name	pg_catalog.text	true
Value	pg_catalog.text	false

Таблиця 2.14 - Таблиця asp_net_user_roles

Ім'я колонки	Тип даних	Not Null
user_id	pg_catalog.int4	true
role_id	pg_catalog.int4	true

Таблиця 2.15 - Таблиця asp_net_user_logins

Ім'я колонки	Тип даних	Not Null
login_provider	pg_catalog.text	true
provider_key	pg_catalog.text	true
user_id	pg_catalog.int4	true
provider_display_name	pg_catalog.text	false

Таблиця 2.16 - Таблиця asp_net_user_claims

Ім'я колонки	Тип даних	Not Null
Id	pg_catalog.int4	true
user_id	pg_catalog.int4	true
claim_type	pg_catalog.text	false
claim_value	pg_catalog.text	false
login_provider	pg_catalog.text	true

2.5 Конструювання програмного забезпечення

2.5.1 Технології при розробці ПЗ

2.5.1.1 Платформа ASP.NET Core

ASP.NET Core – це редизайн ASP.NET 4.x, з архітектурними змінами, що призводить до меншої, більш модульної структури.

ASP.NET Core надає наступні переваги: інтеграція сучасних, клієнтських структур та робочих процесів розробки; хмарна, заснована на середовищі (environment) система конфігурації; вбудована ін'єкція залежності (dependency injection), легкий, високопродуктивний та модульний HTTP-запит; можливість хостингу в IIS, Nginx, Apache, Docker або самостійному хості у власному процесі; можливість створення та запуску у Windows, MacOS та Linux.

ASP.NET Core постачається цілком у вигляді пакетів NuGet. Використовуючи пакети NuGet, можна оптимізувати свій додаток, включивши лише необхідні залежності. Насправді для додатків ASP.NET Core 2.x, націлених на .NET Core, потрібно лише один пакет NuGet. Переваги менших додатків включають безпеку, зменшення обслуговування та покращену продуктивність.

ASP.NET Core MVC надає функції для створення веб-інтерфейсів та веб-програм:

- шаблон Model-View-Controller (MVC) допомагає легше тестувати веб-API та веб-програми;
- Razor Pages – це модель на основі сторінок, що робить створення веб-інтерфейсу легшим та продуктивним;
- розмітка Razor забезпечує продуктивний синтаксис для переглядів Razor Pages та MVC;
- теги Helpers дозволяють включити код сервера для участі у створенні та відтворення елементів HTML у файлах Razor;

- вбудована підтримка декількох форматів даних та узгодження вмісту дозволяє веб-інтерфейсам охопити широкий спектр клієнтів, включаючи веб-переглядачі та мобільні пристрої;
- зв'язок моделі автоматично відображає дані з HTTP-запитів в параметри методів (Model binding);
- валідація моделі автоматично виконує перевірку на стороні клієнта та на стороні сервера;
- вбудовані функції для обробки помилок у додатках, включаючи сторінку винятків розробника, спеціальні сторінки помилок, коди сторінок статичного стану та обробку винятків із запуску;
- надає послуги для локалізації на різних мовах та культурах. [5]

2.5.1.2 Об'єктно-орієнтована мова програмування C#

C# – об'єктно-орієнтована мова програмування. Розроблена в 1998-2001 роках групою інженерів під керівництвом Андерса Хейлсберг в компанії Microsoft як мова розробки додатків для платформи *Microsoft .NET Framework* і внаслідок була стандартизована як ECMA-334 і ISO/IEC 23270.

C# відноситься до сім'ї мов з C-подібним синтаксисом, з них його синтаксис найбільш близький до C++ і Java. Мова має статичну типізацію, підтримує поліморфізм, перевантаження операторів (у тому числі операторів явного і неявного приведення типу), делегати, атрибути, події, властивості, узагальнені типи і методи, ітератори, анонімні функції з підтримкою замикань, *LINQ*, виключення, коментарі у форматі *XML*.

Переїнявши багато що від своїх попередників - мов C++, *Pascal*, Модула, *Smalltalk* і особливо *Java* - C#, спираючись на практику їх використання, виключає деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем, наприклад, C# на відміну від C++ не підтримує множинне спадкування класів (між тим допускалися множинне спадкування інтерфейсів).[7]

2.5.1.3 Структурована мова запитів SQL

SQL створювалася як простий стандартизований спосіб вилучення та управління даними, що містяться в реляційній базі даних. Пізніше він став складніше, ніж замислювався, і перетворився на інструмент розробника, а не кінцевого користувача. В даний час *SQL* залишається найпопулярнішим з мов управління базами даних, хоча й існує ряд альтернатив.

Вихідна назва мови SEQUEL (Structured English Query Language) тільки частково відбиває суть цієї мови. Звичайно, мова була орієнтований головним чином на зручну й зрозумілу користувачам формулювання запитів до реляційних БД. Але, у дійсності, вона майже із самого початку була повною мовою БД, що забезпечує крім засобів формулювання запитів і маніпулювання БД наступні можливості:

- засоби визначення й маніпулювання схемою БД;
- засоби визначення обмежень цілісності й тригерів;
- засоби визначення подань БД;
- засоби визначення структур фізичного рівня, що підтримують ефективно виконання запитів;
- засоби авторизації доступу до відносин й їхніх полів;
- засоби визначення крапок збереження транзакції, і виконання фіксації й відкотів транзакцій. [8]

2.5.1.4 Мова інтегрованих запитів (*LINQ*)

LINQ – компонент Microsoft .NET Framework, який додає можливості виконання запитів даних до мов, що входять у .NET.

LINQ розширює можливості мови, додаючи до неї вирази запитів, що є схожими на твердження *SQL* та можуть бути використані для зручного отримання та обробки даних масивів, XML документів, реляційних баз даних та сторонніх джерел. *LINQ* також визначає набір імен методів (що називаються стандартними операторами запитів, або стандартними операторами послідовностей), а також правила перекладу, що має використовувати компілятор для

					КПІ.ІП-34104.045490.02.81	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

перекладу текучих виразів у звичайні, використовуючи їх назву, лямбда-вирази та анонімні типи.

З появою LINQ, до мов програмування також був додано розширення, щоб полегшити написання запитів:

- неявно типізовані змінні: дане покращення дозволило оголошувати змінні без явного зазначення їх типів. Починаючи з C# 3.0 стало можливим використовувати ключове слово `var`;

- анонімні типи: анонімні типи дозволяють компілятору автоматично створювати та опрацьовувати класи, що містять лише оголошення полів. Це є зручним для таких операторів, як `Select` та `Join`, результат виконання яких часто відрізняється від вхідних типів. Компілятор використовує інтерфейси типів для визначення полів, що мають міститися у вихідному класі і генерує до них методи доступу та зміни для них;

- ініціалізатор об'єкту: ініціалізатори об'єктів дозволяють створювати та ініціалізувати поля об'єкту в одній області видимості, що можна використовувати в операторах `Join` та `Select`;

- лямбда вирази: лямбда вирази дозволяють створювати компактні предикати, або функції проєкції. Вони передаються у функції у вигляді делегатів, або дерев виразів, залежно від провайдера запитів.[9]

2.5.1.5 ASP.NET Core MVC

Архітектурна схема Model-View-Controller (MVC) розділяє програму на три основні групи компонентів: Моделі, Перегляди (Вид) та Контролери. Ця модель допомагає досягти розподілу відповідальності. Згідно цьому шаблону, запити користувачів спрямовуються до Контролера, який відповідає за роботу з Моделлю для виконання дій користувача та/або отримання результатів запитів. Контролер вибирає представлення даних для відображення користувачеві та надає йому необхідні дані моделі.

Цей розподіл відповідальності допомагає масштабувати програму з точки зору складності, оскільки простіше коду

					КПІ.ІП-34104.045490.02.81	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

вати, налагоджувати та протестувати щось (модель, перегляд або контролер), що виконує одну функцію (і слідує принципу єдиної відповідальності). Складніше оновити, протестувати та налагоджувати код, який має розповсюдженні залежності між двома або більше з цих трьох областей. Наприклад, логіка інтерфейсу користувача змінюється частіше, ніж бізнес-логіка. Якщо код представлення та бізнес-логіка об'єднуються в єдиному об'єкті, об'єкт, що містить бізнес-логіку, повинен бути змінений кожного разу, коли змінюється інтерфейс користувача. Це часто вводить помилки і вимагає повторної перевірки бізнес-логіки після кожного мінімального зміни інтерфейсу користувача.

І вид, і контролер залежать від моделі. Однак модель не залежить ні від перегляду, ані від контролера. Це одна з основних переваг поділу. Це розділення дозволяє побудувати та перевіряти модель незалежно від візуального представлення.

ASP.NET Core MVC надає заснований на шаблонах спосіб побудови динамічних веб-сайтів, що дозволяє точно розділити рішення (відповідальність). Це дає повний контроль над розміткою, підтримує TDD-дружній розвиток та використовує найновіші веб-стандарти.

Одна з найголовніших переваг – ASP.NET Core має вбудовану підтримку ін'єкцій залежності (DI). У ASP.NET Core MVC, контролери можуть запитувати необхідні сервіси через свої конструктори, дозволяючи їм слідувати принципу явних залежностей.

Перегляди ASP.NET Core MVC використовують движок перегляду Razor для відтворення переглядів. Razor – це компактна мова розмітки шаблонів для визначення переглядів за допомогою вбудованого коду C#. Razor використовується для динамічного створення веб-контенту на сервері, завдяки цьому можна змішувати серверний код із кодом сторони клієнта. Також підтримуються сильно типізовані вигляди (Strongly typed views), коли можна передати певний тип моделі і радіти всім перевагам IntelliSense, який підказує і перевіряє.[6]

2.5.1.6 React

React – відкрита JavaScript бібліотека для створення інтерфейсів користувача, яка покликана вирішувати проблеми часткового оновлення вмісту веб-сторінки, з якими стикаються в розробці односторінкових застосунків. Розробляється Facebook, Instagram і спільнотою індивідуальних розробників.

React дозволяє розробникам створювати великі веб-застосунки, які використовують дані, котрі змінюються з часом, без перезавантаження сторінки. Його мета полягає в тому, щоб бути швидким, простим, масштабованим. React обробляє тільки користувацький інтерфейс у застосунках. Це відповідає видові у шаблоні модель-вид-контролер (MVC), і може бути використане у поєднанні з іншими JavaScript бібліотеками або в великих фреймворках MVC, таких як AngularJS. Він також може бути використаний з React на основі надбудов, щоб піклуватися про частини без користувацького інтерфейсу побудови веб-застосунків. Як бібліотеку інтерфейсу користувача React найчастіше використовують разом з іншими бібліотеками, такими як Redux.

В даний час React використовують Khan Academy, Netflix, Yahoo, Airbnb, Sony, Atlassian та інші. Бібліотеку створено Джорданом Волком (Jordan Walke), програмістом з Facebook. Автор працював над проектом під впливом ХНР, фреймворку HTML для PHP. 2011-го року реліз з'явився у новинах Facebook, за рік — у блозі Instagram. Також фреймворк був представлений як проект з відкритим початковим кодом на конференції розробників JSConf US, що проходила у Сполучених Штатах у травні 2013 року. На конференції React.js Conf, влаштовану Фейсбуком у березні 2015-го, проект було представлено як відкрите програмне забезпечення. [10]

					КПІ.ІП-34104.045490.02.81	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

2.6 Висновки по розділу

У розділі 2 наведено архітектуру програмного забезпечення, детально розібрано підходи до розробки, описано класи предметної області, наведено схему бази даних та її опис. Було коротко оглянуто технології, що використовувались при розробці ПЗ.

Розроблено програмне забезпечення, використовуючи сучасні технології.

					КПІ.ІП-34104.045490.02.81	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Аналіз якості ПЗ

Тестування є важливою частиною процесу розробки програмного забезпечення. У випадку розробки під мобільну і десктопну версію екрану тестування має особливу важливість, оскільки використовувані пристрої досить сильно відрізняються один від одного за розміром і розміром екрану.

Програмне забезпечення обов'язково тестується на декількох пристроях. Значна кількість дефектів не залежить від оточення, тобто вони відтворюються на всіх пристроях, але є випадки, коли дефект можливо відтворити лише на одному пристрої.

Користувачі програмного забезпечення очікують високої якості від відвідуваного сайту, його якість безпосередньо впливає на відвідуваність, наскільки буде популярним сайт і як довго він буде відвідуватись.

Первинне тестування веб-ресурсу.

Повний тест передбачає перевірку всієї функціональності програми:

- перевірки по позитивним і негативним сценаріям;
- взаємодія функцій програми і призначеного для користувача інтерфейсу;
- специфічні перевірки для мобільних пристроїв і комп'ютерів.

Тест сумісності включає перевірку тільки основної функціональності програмного забезпечення, а також відображення на екранах пристроїв різного розміру. Даний тест включає тільки позитивні сценарії і не передбачає перевірок нестандартних варіантів використання програми.

3.2 Опис процесів тестування

Для тестування були використані пристрої з різним розміром екрану. Серед них усі доступні симулятори пристроїв в *Google Chrome*. Дані процедури

					КПІ.ІП-34104.045490.02.81	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

тестування допомагають виявити різні типи проблем з якістю програмного забезпечення.

3.3 Опис контрольного прикладу

В процесі тестування була перевірена уся функціональність системи.

Послідовно були перевірені всі варіанти використання, результати представлені у відповідних таблицях:

- автентифікація (Таблиця 3.1);
- реєстрація (Таблиця 3.2);
- створення повідомлення про шлюбний політ (Таблиця 3.3);
- видалення повідомлення (Таблиця 3.4);
- редагування повідомлення (Таблиця 3.5);
- перегляд тенденцій шлюбного польоту (Таблиця 3.6);
- верифікація шлюбного польоту (Таблиця 3.7);
- відмова у верифікації шлюбного польоту (Таблиця 3.8).

Таблиця 3.1 – Перевірка можливості автентифікації

Мета тесту	Перевірка можливості автентифікації
Початковий стан	Відкрита сторінка входу
Вхідні дані	Логін і пароль профілю користувача
Схема проведення тесту	Заповнити поле логін і пароль. Натиснути кнопку «Войти».
Очікуваний результат	Користувача авторизовано.
Стан програмного продукту після проведення випробувань	Авторизація пройшла успішно. Відкрита головна сторінка.

Таблиця 3.2 – Перевірка можливості реєстрації

Мета тесту	Перевірка можливості автентифікації
Початковий стан	Відкрита сторінка реєстрації

Вхідні дані	Електронна пошта і пароль профілю користувача
Схема проведення тесту	Заповнити поле email і пароль. Натиснути кнопку «Зареєструватися».
Очікуваний результат	Користувача зареєстровано.
Стан програмного продукту після проведення випробувань	Реєстрація пройшла успішно. Відкрита головна сторінка.

Таблиця 3.3 – Перевірка можливості створення повідомлення про політ

Мета тесту	Перевірка можливості створення повідомлення про політ
Початковий стан	Відкрита головна сторінка
Вхідні дані	Опис повідомлення про шлюбний політ, локація польоту, дата польоту, вид.
Схема проведення тесту	Натиснути на кнопку «Мої повідомлення». Натиснути на кнопку «Створити». Заповнити поля вид, дата польоту, опис. Натиснути кнопку «Створити політ».
Очікуваний результат	Створено повідомлення про політ.
Стан програмного продукту після проведення випробувань	Відкрита сторінка мої повідомлення. Коротка інформація верхнього повідомлення відповідає заповненій інформації полям.

Таблиця 3.4 – Перевірка можливості видалення повідомлення про політ

Мета тесту	Перевірка можливості видалення повідомлення про політ
------------	-------------------------------------------------------

Початковий стан	Відкрита головна сторінка
Схема проведення тесту	Натиснути на кнопку «Мої повідомлення». Натиснути на кнопку видалити повідомлення.
Очікуваний результат	Повідомлення видалено
Стан програмного продукту після проведення випробувань	Відкрита сторінка мої повідомлення. Повідомлення, напроти якого натиснута кнопка зникло.

Таблиця 3.5 – Перевірка можливості редагування повідомлення про політ

Мета тесту	Перевірка можливості редагування повідомлення про політ
Початковий стан	Відкрита головна сторінка
Вхідні дані	Опис повідомлення про шлюбний політ, локація польоту, дата польоту, вид.
Схема проведення тесту	Натиснути на кнопку «Мої повідомлення». Обрати повідомлення та натиснути «Редагувати». Заповнити поля вид, дата польоту, опис. Натиснути кнопку «Редагувати».
Очікуваний результат	Повідомлення змінено.
Стан програмного продукту після проведення випробувань	Відкрита сторінка мої повідомлення. Коротка інформація верхнього повідомлення відповідає заповненій інформації полям. Статус повідомлення - Змінено.

Таблиця 3.6 – Перевірка можливості перегляду тенденцій польоту

Мета тесту	Перевірка можливості перегляду тенденцій польоту.
------------	---------------------------------------------------

Початковий стан	Відкрита головна сторінка
Вхідні дані	Таксон , локація.
Схема проведення тесту	Ввести таксон і локацію.
Очікуваний результат	Графік змінився
Стан програмного продукту після проведення випробувань	Відкрита головна сторінка

Таблиця 3.7 – Перевірка можливості верифікації шлюбного польоту

Мета тесту	Перевірка можливості верифікації повідомлення про шлюбний політ
Початковий стан	Відкрита головна сторінка
Вхідні дані	
Схема проведення тесту	Натиснути на кнопку Очікують верифікації. Обрати повідомлення та натиснути на кнопку «Верифікувати», буде відкрита сторінка повідомлення. Натиснути на кнопку «Верифікувати».
Очікуваний результат	Повідомлення верифіковане
Стан програмного продукту після проведення випробувань	Відкрита сторінка повідомлення що очікують верифікацію, повідомлення що було верифіковано – відсутнє.

Таблиця 3.8 – Перевірка можливості верифікації шлюбного польоту

Мета тесту	Перевірка можливості верифікації повідомлення про шлюбний політ
Початковий стан	Відкрита головна сторінка
Вхідні дані	Причина відмови.

Продовження таблиці 3.8

Схема проведення тесту	Натиснути на кнопку «Очікують верифікації». Обрати повідомлення та
------------------------	--------------------------------------------------------------------

	натиснути на кнопку «Верифікувати», буле відкрита сторінка повідомлення. На даній сторінці ввести причину відмови. Натиснути кнопку «Відмовити».
Очікуваний результат	Повідомлення відмовлено у верифікації
Стан програмного продукту після проведення випробувань	Відкрита сторінка повідомлення що очікують верифікацію, повідомлення, якому відмовили у верифікації, – відсутнє.

3.4 Висновки по розділу

У даному розділі був створений тестовий план, та проведене тестування основних функцій додатку, таких, як:

- автентифікація ;
- реєстрація;
- створення повідомлення про шлюбний політ;
- видалення повідомлення ;
- редагування повідомлення;
- перегляд тенденцій шлюбного польоту;
- верифікація шлюбного польоту;
- відмова у верифікації шлюбного польоту.

Під час процесу тестування усі тести пройшли успішно, що свідчить про відмінну якість продукту.

4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Розгортання програмного забезпечення

Програмне забезпечення може бути розгорнуте на персональних комп'ютерах або на серверах.

4.1.1 Мінімальна конфігурація технічних засобів

Тип процесору: процесор x86 з тактовою частотою 1,0 ГГц або

процесор x64 з тактовою частотою 1,4 ГГц

Об'єм ОЗП: 1Гб.

Об'єм фізичної пам'яті: 64Гб.

4.1.2 Вимоги до операційних систем

Операційна система: Windows (7+, Server8+), macOS (10.12+ «Sierra»), Linux (Ubuntu14.04+, Fedora 26+, Debian 8.7+)

4.1.3 Вимоги до інформаційної та програмної сумісності

Для того, щоб зайти на сайт необхідно мати браузер на пристрої, наприклад Chrome, Opera, Firefox, IE, EDGE, Safari.

4.1.4 Робота з програмним забезпеченням

Керівництво користувача наведено в документі *КПІ.ІП-34104.045490.05.34 «Керівництво користувача»*.

					КПІ.ІП-34104.045490.02.81	Арк.
						49
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Була проаналізована предметна область, виявлені проблеми, проаналізовано існуючі рішення, виділено функціональні та нефункціональні вимоги, спроектовано та розроблено відповідне програмне забезпечення.

Проект вирішує проблему розрізненості знаходження даних щодо шлюбного польоту мурах та як наслідок складності визначення тенденцій його календарних рамок. Всі функціональні вимоги було виконано і протестовано.

Було вивчено, засвоєно та використано:

- принцип роботи MVC;
- принцип роботи CodeFirst, EntityFramework Core та міграцій;
- принципи побудови адаптивного дизайну;
- використання Razor двигуна;
- використання React;
- розробка технічної документації за ДСТУ.

					КПІ.ІП-34104.045490.02.81	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ПОСИЛАНЬ

- 1) Форум мірмікіперів [Електронний ресурс] Режим доступу : <https://antclub.org>
- 2) Форум мірмікіперів [Електронний ресурс] Режим доступу : <https://antclub.ru>
- 3) Північно американський форум мірмікіперів [Електронний ресурс] Режим доступу : <https://www.formiculture.com>
- 4) Німецький форум мірмікіперів [Електронний ресурс] Режим доступу : <https://www.ameisenforum.de>
- 5) Огляд ASP.NET Core [Електронний ресурс] Режим доступу: <https://docs.microsoft.com/en-us/aspnet/core/?view=aspnetcore-2.1>
- 6) Огляд ASP.NET Core MVC [Електронний ресурс] Режим доступу: <https://docs.microsoft.com/en-us/aspnet/core/mvc/overview?view=aspnetcore-2.1>
- 7) Огляд C# [Електронний ресурс] Режим доступу: https://ru.wikipedia.org/wiki/C_Sharp
- 8) Огляд SQL [Електронний ресурс] Режим доступу: <https://www.ua5.org/osnprog/111-storja-movi-sql.html>
- 9) Огляд LINQ [Електронний ресурс] Режим доступу: <https://uk.wikipedia.org/wiki/LINQ>
- 10) Огляд React [Електронний ресурс] Режим доступу: <https://uk.wikipedia.org/wiki/React>
- 11) Сторінка мірмікіперів NebraskaAnts у соціальній мережі [Електронний ресурс] Режим доступу: <https://www.facebook.com/NebraskaAnts>